An Augmented Learning Approach for Multiple Data Streams Under Concept Drift

Kun Wang[1[0000-0003-2711-6233]], Jie Lu[1[0000-0003-0690-4732]], Anjin Liu[1[0000-0002-0733-7138]], and Guangquan Zhang[1[0000-0003-3960-0583]]

Australian Artificial Intelligence Institute, University of Technology Sydney, Australia
{Kun.Wang, Jie.Lu, Anjin.Liu, Guangquan.Zhang}@uts.edu.au

Abstract. Multiple data stream learning attracts more and more attention recently. Different from learning a single data stream, the uncertain and complex occurrence of concept drift in multiple data streams, bring challenges in real-time learning performance. To address this issue, this paper proposed a method called time-warping-based concept drift learning method (TW-CDM) for dealing with multiple data streams. First, a time-warping-based drift identification process is given to recognize the drift region. Second, an augmented learning process is developed by crossly using the located region data. Finally, a selectively augmented learning process is given to reduce the influence of different drift severity. The proposed method is evaluated on both synthetic and real-world datasets, and compared with benchmark methods. The experiment results show the efficiency of the proposed method.

Keywords: Concept drift · Data stream · Multiple streams · Ensemble learning

1 Introduction

Data streams commonly exist in our real-world life and attract high research attention in recent years [17,9]. Scenarios like weather changes, price fluctuates, and user interest drifts are representative of streaming data. Related research about handling concept drift for data stream learning has got successful progress. However, most of the previous work aims to address the concept drift learning in a single stream, and few of them consider the situation that deals with multiple data streams [27,25]. On the one hand, multiple data stream learning is relatively more complex, since the concept drift situations in each of them are different. Maintaining the learning performance on such complex task is a challenge. On the other hand, with the higher requirement of real-world prediction, learning tasks on multiple data streams need to be enhanced for better application. Therefore, it is necessary to consider an efficient method to handle concept drift for multiple data stream learning.

Recently, there are related works that focus on multiple data stream learning and develop several markable outputs. We found that there are three main task settings of multiple data stream learning: Supervised [25,28] and unsupervised [26,24,22] tasks. These previously proposed approaches provide optimization and performance improvement of learning strategies for processing multiple data streams. However, few of them consider identifying and adapting different situations of concept drift in such a complex scenario.

Learning concept drift in multiple data streams is more complex than handling it in a single data stream. First, in multiple data streams, concept drift may occur asynchronously and will lead to model learning decay [27,5]. Second, the drift severity in multiple data streams may also change differently, which will interfere with model learning. Third, there may be correlation and interaction between multiple data streams, which brings difficulties to model learning. Besides, learning efficiency by selecting appropriate knowledge to deal with multiple streams should also be addressed. Therefore, learning multiple data streams under concept drift is a challenge that should be highly addressed.

Motivated by this, we aim to give a clear definition of concept drift in multiple data streams, and try to find an appropriate learning solution. To address these issues, this paper proposes a time-warping-based multiple data stream learning method (TW-CDM) to deal with concept drift adaptatively. The contribution of this paper is shown below:

- A time warping-based strategy has been given to help identify the possible drifted data in multiple data streams. This process can help not only recognize the changeable data, but also reflect the drift severity in time.
- An augmented learning approach has been developed to help adapt to the concept drift that occurs on multiple data streams. The update process is triggered and augmented by the results of the drift-identified process.
- The experiment on multiple data streams with different drift situations shows the effectiveness of the proposed method. And the source code is available online [1].

The rest of this paper is organized as follows. Section 2 summarizes the recent works. The research problem statement and the proposed approaches are introduced in Section 3. And Section 4 describes the experiment setting, datasets, and benchmark methods. The experiment results are discussed in Section 5. Finally, a summary of the conclusion and our future study is given in Section 6.

2 Related Work

In this section, we give an overview of research works about multiple data stream learning and concept drift learning.

2.1 Multiple Data Stream Learning

Recently, many research works focusing on multiple data stream learning based on different task settings. To summarize, current studies of multiple data stream learning are mainly based on the settings of supervised and unsupervised tasks. For supervised tasks, research [27] addresses the issue of concept drift in multiple data streams, and a group concept drift detection method has been proposed to support model learning. In

[1] https://github.com/kunkun111/TW-CDM

addition, a method called fuzzy drift variance (FDV) method has been given to measure the correlated drift patterns among multiple data streams [25]. Aiming to handle multiple relevant data streams, a method called MuNet has been developed to improve learning efficiency by leveraging the dependency among multiple data streams [28]. To address the issue of sequential pattern mining, research [11] proposes a method called PSP-AMS method to help better identify the patterns in multiple data streams.

For unsupervised tasks, to deal with the concept evolving nature of data streams, research [24] proposes a fuzzy hierarchical clustering method for clustering multiple nominal data streams. Besides, a dynamic ensemble clustering algorithm based on evidence accumulation has been generated to address missing data and delayed data in multiple data streams [23]. Also, the data stream clustering method based on multiple medoids and medoid voting has been proposed to improve the efficiency of sequence clustering [22]. A histogram-based clustering method has been proposed to deal with online multiple data streams [3]. Besides, focusing on the problem of unlabelled drifting streams, a framework called Learn-to-Adapt (L2A) has been developed to handle concept drift adaptation tasks in multiple data streams [26]. Furthermore, research [5,10] handle multiple data streams according to the idea of transfer learning [20] and domain adaptation [7].

In addition, it is also needed to consider real-world decision-making [18]. Application tasks like multimodal learning [21] and multi-source learning [19] may also face the impact of uncertainty.

2.2 Concept Drift Learning Under Uncertainty

Concept drift learning is an interesting topic that recently gets highly focused deal with the impact of the uncertain environment [17,9,13]. With the issue that the changeable distribution of streaming data may cause learning decay and performance degradation, many research works aim to detect, understand, and adapt to concept drift.

For concept drift detection, commonly used methods include DDM [8], EDDM [2], ADWIN [4], and so on. Most drift detection methods identify changeable data based on drift threshold, error rate, and statistic results. For concept drift understanding, research works [16,15] introduce the strategy to locate the region of concept drift to support better understanding. This idea not only can identify where the drift occurs, but also can help update the learning model based on how drift occurs. For concept drift, the main target is to figure out the appropriate strategy to help the learning model adapt to the changeable data stream. For example, Learn++.NSE [6] as a classical drift learning method, main the performance based on ensemble learning and dynamically weighted majority voting. Also, the method of dynamic weighted majority [12] has been proposed to help handle data stream with concept drift. Recently, novelty methods have also been proposed to enhance concept drift learning, like adaptative minimax risk classifiers (AMRCs) [1], and data distribution generation for predictable concept drift adaptation (DDG-DA) [14].

However, most of the current works aim to deal with concept drift occurring on a single data stream, few of them consider the scenario when drift occurs on multiple data streams asynchronously, which is a relatively more complex learning task. The same as learning a single data stream, drift identification and learning also plays an important role. Therefore, the target of this paper is to figure out the strategy that can ensure the machine learning model can deal with multiple data streams occurs concept drift.

3 Methodology

This section first gives the problem description of concept drift in multiple data streams, and then develops a time warping-based model for multi-stream learning to help detect and adapt to concept drift.

3.1 Problem Statement

In our problem setting, we initially focus on dealing with two data streams with the same size and homogeneous structures, but different concept drift situations. Given two data streams $S_1 = \{x_i^{S_1}, y_i^{S_1}\}_{i=1}^n$, $S_2 = \{x_i^{S_2}, y_i^{S_2}\}_{i=1}^n$, follow the distribution P, Q. As time goes on, the data distribution of these two streams may change from time t to time $t+1$, causing different concept drift situations, leading to learning model degradation. Here, we give a definition of concept drift in multiple data streams as below.

Definition 1. *(Concept Drift in Multiple Data Streams) For data streams S_1, S_2, concept drift occurs when the data distribution of data streams changes, denote as*

$$P_t\left(y^{S_1}|x^{S_1}\right) \neq P_{t+1}\left(y^{S_1}|x^{S_1}\right) \vee Q_t\left(y^{S_2}|x^{S_2}\right) \neq Q_{t+1}\left(y^{S_2}|x^{S_2}\right), \tag{1}$$

where P and Q are different distributions followed by streams S_1 and S_2.

When concept drift occurs in multiple data streams, the machine learning model should real-time identify and adapt this uncertain change in both of them. To deal with this complex scenario, it is needed to find an appropriate method $\mathcal{F}$ to help reduce the model cost L when dealing with multiple data streams with concept drift. Therefore, our research target is

$$\mathcal{F} = \arg\min_{\mathcal{F}} \sum_{t=1}^{T} L_t\left(\mathcal{F}\left(\left\{x^{S_1}, x^{S_2}\right\}\right), \left\{y^{S_1}, y^{S_2}\right\}\right). \tag{2}$$

Aiming to address this issue, the following paper develops drift identification and adaptation process for multiple data stream learning.

3.2 Time Warping-based Drift Identification Process

In this section, we first consider real-time drift identification in multiple data streams. We train a model $f(x)$ on chunks $D_t^{S_1}$, $D_t^{S_2}$, and then test it on chunks $D_{t+1}^{S_1}$, $D_{t+1}^{S_2}$. Get the prediction results and calculate the prequential accuracy $\varepsilon_{t+1}^{S_1}$, $\varepsilon_{t+1}^{S_2}$ of two streams.

$$\varepsilon_{t+1}^{S_1} = \left\{\varepsilon_{t+1,i}^{S_1}\right\}_{i=1}^{n}, \ \ \varepsilon_{t+1}^{S_2} = \left\{\varepsilon_{t+1,i}^{S_2}\right\}_{i=1}^{n} \tag{3}$$

Motivated the by dynamic time-warping method, which can measure the similarity of two series data. It can help identify the change and differences between two series of

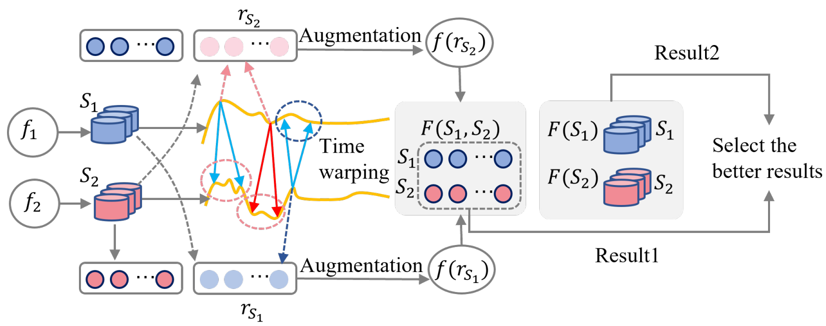


Fig. 1. An illustration of the framework of the time-warping-based concept drift learning method for multiple data stream.

data. Motivated by this, we apply this method to calculate the distance between these two series, denote as

$$d\left(\varepsilon_{t+1}^{S_1}, \varepsilon_{t+1}^{S_2}\right) = \left|\varepsilon_{t+1}^{S_1} - \varepsilon_{t+1}^{S_2}\right| \tag{4}$$

Thus, the matching path of these two series can be got. The matching path means that the parts of the two sequences are similar.

$$\varepsilon_{t+1,i}^{S_1} \leftarrow \varepsilon_{t+1,j}^{S_2} \text{ or } \varepsilon_{t+1,i}^{S_1} \leftarrow \left\{\varepsilon_{t+1,j}^{S_2}, \cdots, \varepsilon_{t+1,j+\eta}^{S_2}\right\} \tag{5}$$

And we choose the case of $\varepsilon_{t+1,i}^{S_1} \leftarrow \left\{\varepsilon_{t+1,j}^{S_2}, \cdots, \varepsilon_{t+1,j+\eta}^{S_2}\right\}$, and then find the degraded accuracy, and identify drift that occurs in S_1, locate the regions of possible drift occurrence in S_2.

3.3 Learning Augmentation for Drift Adaptation

In this section, we try to use the identified regions of data to help support the model learning. Since the results of time warping reflect that there is similarity between S_1 and S_2 at some time steps. So, we select the data of two streams crossly to augment the learning process. First, merging the chunks at time t as a new train set, denote as

$$D_t = D_t^{S_1} \cup D_t^{S_2} \tag{6}$$

Then, train a based model $F(x)$ at time t and test it on $D_{t+1}^{S_1}$, $D_{t+1}^{S_2}$ at time $t+1$, and identified changeable region of two streams, r_{S_1}, r_{S_2}. Then update the train set at current time point for the base model learning as

$$D_{t+1} = \left\{D_{t+1}^{S_1}\right\} \cup \left\{D_{t+1}^{S_2}\right\} \tag{7}$$

Finally, retrain the based model on the updated train set, denote as $F(D_{t+1})$. Then separately train a single model on r_{S_1}, r_{S_2}, denote as $f(r_{S_1})$, $f(r_{S_2})$. Let λ is a parameter, the updated model for predicting two data streams can be expressed as

$$\hat{y}^{S_1} = F(D_{t+1}) + \lambda f(r_{S_2}), \quad \hat{y}^{S_2} = F(D_{t+1}) + \lambda f(r_{S_1}), \tag{8}$$

Algorithm 1: Time-Waping-based Concept Drift Learning Method

Input : Data chunks of two data streams D^{S_1}, D^{S_2}.
Output: Model prediction results of two streams.
Begin
Separately train a model $f(x)$ on chunks $D_1^{S_1}$, $D_1^{S_2}$ at time $t = 1$;
for $t = 2$ **to** T **do**
  Test model on chunks $D_t^{S_1}$, $D_t^{S_2}$ and get prediction results;
  Calculate the prequential accuracy $\varepsilon_t^{S_1}$, $\varepsilon_t^{S_2}$ of two streams;
  Calculate the distance between two streams by Eq. (4);
  Get the matching path of two streams by Eq. (5);
  Recognize the changeable regions r_{S_1}, r_{S_2} by Eq. (7);
  if *No changeable regions* **then**
    Update the train set by Eq. (6);
    Retrain base model $F(x)$;
  end
  if *Identify changeable regions* **then**
    Update the train set by Eq. (7);
    Retrain base model $F(x)$, train tree models for two streams;
    Get the prediction result by Eq. (8);
    Retrain base model $F(x)$;
    Get the prediction result by Eq. (9);
  end
  Choose the better prediction results by Eq. (10).
end
End

3.4 Selectively Augmented Learning Process

In this section, we consider different drift situations and aim to maintain the learning performance. We selectively use the augmented learning process and initial retraining process to reduce the influence of the difference between the two streams. So, we initially retrain the based model on the updated training set and output results, denote as

$$\hat{y}'^{S_1} = F(D_{t+1}), \;\; \hat{y}'^{S_2} = F(D_{t+1}), \tag{9}$$

Considering the uncertainty, we selectively choose the better results after all the training processes have been finished, and the results have been output, denote as

$$\min\left(L_{Aug}\left(\left\{y^{S_1}, y^{S_2}\right\}, \left\{\hat{y}^{S_1}, \hat{y}^{S_2}\right\}\right), L_{Ini}\left(\left\{y^{S_1}, y^{S_2}\right\}, \left\{\hat{y}'^{S_1}, \hat{y}'^{S_2}\right\}\right)\right) \tag{10}$$

Moreover, this proposed method is currently deals with two parallel streams, it can also be extended to handle a larger number of data streams, which would be our future research target.

4 Experiment

This section introduces the experiment settings, synthetic and real-world datasets description, and experiment results discussion.

4.1 Experiment Setting

In this experiment, we test and evaluate the proposed method on the datasets, then output and discuss the results. First, for the model usage, a gradient boosting decision tree model has been used as the base model, and a decision tree model has been used for learning the augmented data. Both of them are from the Sklearn package with default parameter setting. To reduce the influence of the randomness, the random state of the learning model is set as 0. Both of them are from the sklearn package with default parameter settings. Second, the learning process is data chunk-based and the prequential train and test principle is applied. Moreover, the accuracy and F1-score metrics are chosen for the results evaluation. There are three main steps of the experiment, and the detailed procedures are listed as follows:

- **Step 1:** To help recognize the drift that occurs on multiple data streams, we test the proposed time warping-based drift identification process on data chunks and then record the drift frequency.
- **Step 2:** To reduce the influence of drifted data, test the proposed drift adaptation method which embedded the data augmentation process on multiple data streams, and then discuss the experiment results.
- **Step 3:** To maintain the learning performance, we selectively choose the traditional retrain and augmented retrain processes to help adapt to multiple data streams with various drift situations.

4.2 Datasets

In the experiment, we choose both synthetic and real-world datasets to evaluate the proposed method. Since this is our first attempt to deal with multiple data streams, the experiment setting is learning and testing the proposed method on two data streams of the same size at the same time. And the chunk size is set as 100 default. Therefore, we design different scenarios to simulate multiple data stream learning. For the synthetic datasets, we generate six data streams with different drift situations. Each of them contains 10,000 data instances. We simulate the drift occurrence by rotating the decision boundary, the degree of the rotating process is θ. Thus, different drift types have been simulated, such as sudden drift, incremental drift, and a mixture of both of them. drift types have been simulated, such as sudden drift, incremental drift, and a mixture of both of them. Then, we use the first data stream as the base stream, and handle this stream with one of the other streams. For the real-world datasets, we first segment the dataset into two data streams, then learn and test the proposed method on them.

- **Stream 1:** is the base data stream which is generated with sudden drift from the 3,001-st to the 6,000-th time points by changing the decision boundary.
- **Stream 2:** also is generated with sudden drift. The drift severity is the same as Stream 1, but the drift time point is late, and the drift period is relatively longer, which is from the 4,001-st to the 8,000-th time point.
- **Stream 3:** simulates a higher frequency of sudden drift, the decision boundary changes every 2,000 time points with the same drift severity as Stream 1.

Table 1. Datasets Description

Datasets	Instances	Attributes	Classes	Chunk size	Drift type	θ value
Stream 1	10,000	3	2	100	Sudden(short)	0,0,180,180,180,180,0,0,0,0
Stream 2	10,000	3	2	100	Sudden(long)	0,0,0,180,180,180,180,180,0,0
Stream 3	10,000	3	2	100	Sudden(high)	0,0,180,180,0,0,180,180,0,0
Stream 4	10,000	3	2	100	Sudden(low)	0,0,90,90,180,180,-90,-90,0,0
Stream 5	10,000	3	2	100	Incremental	0,45,90,135,180,-135,-90,-45,0,0
Stream 6	10,000	3	2	100	Both	0,45,90,135,180,180,0,0,0,0
Weather	25,626	8	2	365	-	-
Electricity	45,312	8	2	100	-	-

- **Stream 4:** simulates the same drift frequency as Stream 3, that is, the decision boundary changes every 2,000 time points with a relatively lower drift severity.
- **Stream 5:** generates the data occurs with incremental drift, the decision boundary changes every 1,000 time points with a lower drift severity.
- **Stream 6:** contains data with incremental and sudden drift, incremental drift occurs from 1-st to the 5,000-th time point while sudden drift at the 7,001-st time point.
- **Weather:** is the record data collected from the NOAA datasets. This data contains 25,626 data instances, 8 attributes, and 2 classes. The chunk size of this data is set as 365. We segment this data into two sets with the same size for the experiment.
- **Electricity:** contains 8 attributes and 2 classes, and 45,312 data instances. We segment this data into two sets with the same size for the experiment.

The prediction results will be evaluated by accuracy and f1-score. The detailed information of each dataset is shown in Table 1.

4.3 Benchmark Methods

To evaluate the performance of the proposed method on multiple data streams, several benchmark methods have been chosen for comparison. The detailed information on each benchmark method is listed below:

- **Baseline:** combines the instances of multiple data streams in the first time step, and initially trains a based model. Then, testing the model in the data chunk of followed time steps without any update.
- **Retrain:** is a strategy that initially trains a base model on streaming data at the first time step, then tests and retrains it on the new incoming chunks.
- **Augmented learning (AL) process:** is embedded with the data augmentation process to help learning multiple data streams. Data selected by the time warping process have been used in the learning process for augmented learning.
- **Selectively augmented learning (TW-CDM) process:** considers the situation that there is a significant distribution difference in multiple data streams. This method aims to selectively use the initial retrain and augmented retrain method to maintain the learning performance.

Table 2. Results of time-warping-based drift identification process

Tasks	$S_1 \rightarrow S_2$	$S_1 \rightarrow S_3$	$S_1 \rightarrow S_4$	$S_1 \rightarrow S_5$	$S_1 \rightarrow S_6$
Change frequency	19	26	29	33	21

Table 3. Friedman test results with benchmark methods

Methods	Baseline	Retrain	AL	TW-CDM$_{\lambda=0.05}$
p value	$0.00815 < 0.01$	$0.00815 < 0.01$	$0.00815 < 0.01$	$0.05878 > 0.01$

Python 3.7 was used to implement the proposed method. The computational environment is a Red Hat Enterprise Linux Workstation release 7.9 (Maipo), with Intel(R) Xeon(R) Gold 6238R CPU @ 2.20GHz.

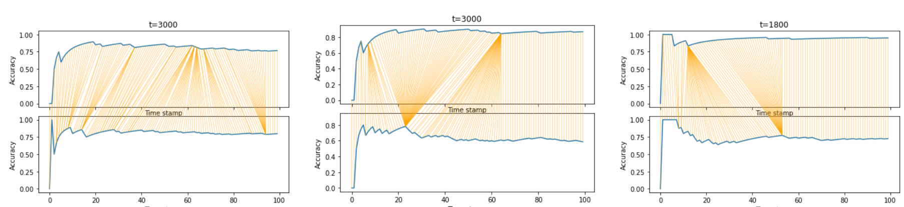
5 Experiment Discussion

This section gives a detailed discussion of the experiment results.

5.1 Evaluate the Drift Identification Process

In this experiment, we test the proposed time warping-based drift identification process on both synthetic and real-world datasets, and then record the drift frequency. As shown in Table 2, this process can not only help recognize the drift occurrence in time, but also reflect the drift severity according to the recorded frequency.

Different from some traditional drift detection methods which are based on a manually set drift threshold, the proposed method try to find the similar part of two data streams based on the model prequential accuracy on two streams' data chunks as time series. Thus, not only the change can be found easily, but also the regions in which has the same similarity and possibly occurs change can also be located.



(a) $S_1 \rightarrow S_2$, t=3000 (b) $S_1 \rightarrow S_4$, t=3000 (c) $S_1 \rightarrow S_5$, t=1800

Fig. 2. A plot of the time-warping results of different multiple data stream scenarios.

Table 4. Average chunk accuracy of time-warping-based concept drift learning method (%)

Tasks	$S_1 \to S_2$	$S_1 \to S_3$	$S_1 \to S_4$	$S_1 \to S_5$	$S_1 \to S_6$	Weather	Electricity
Baseline	49.52	59.59	59.12	59.79	64.55	68.51	67.06
Retrain	84.18	77.31	87.61	86.40	91.92	77.85	76.00
AL	83.15	74.79	86.72	85.58	91.50	77.56	75.56
TW-CDM$_{\lambda=0.05}$	84.87	77.40	87.66	86.49	91.97	78.14	76.42
TW-CDM$_{\lambda=0.1}$	84.98	77.51	87.69	86.54	91.99	**78.31**	76.55
TW-CDM$_{\lambda=0.5}$	85.38	**77.97**	**87.95**	**86.66**	92.03	77.94	77.73
TW-CDM$_{\lambda=0.8}$	**85.39**	77.95	87.92	86.64	**92.03**	77.97	**77.76**

Table 5. Average chunk f1-score of time-warping-based concept drift learning method (%)

Tasks	$S_1 \to S_2$	$S_1 \to S_3$	$S_1 \to S_4$	$S_1 \to S_5$	$S_1 \to S_6$	Weather	Electricity
Baseline	49.45	59.63	59.12	59.52	64.53	72.16	60.84
Retrain	84.95	76.75	87.78	86.12	91.67	82.63	76.42
AL	84.40	75.84	87.20	85.60	91.40	82.97	76.78
TW-CDM$_{\lambda=0.05}$	85.09	76.89	87.85	86.24	91.73	83.04	77.07
TW-CDM$_{\lambda=0.1}$	85.28	77.05	87.89	86.30	91.77	**83.29**	77.32
TW-CDM$_{\lambda=0.5}$	85.79	**77.59**	**88.16**	86.47	**91.80**	82.92	**79.78**
TW-CDM$_{\lambda=0.8}$	**85.80**	77.57	88.14	**86.49**	91.79	82.95	79.76

5.2 Measure the Augmented Learning Process

In this experiment, we use the data in the located region to help model learning. And we evaluate the proposed augmented training process by testing it on both synthetic and real-world datasets, then calculate the accuracy and f1-score, the results are shown in Tables 4, 5. By comparing with the baseline, the proposed augmented training process got a relatively higher evaluation score. And the data streams that occur sudden/incremental drift can be handled with relatively higher performance. Furthermore, the located changeable regions help the model learning and adaptation.

There are both advantages and disadvantages to this method. According to the experiment results, the data in the located region from other data stream can help increase the learning ability of the base model. That is to say, the knowledge in multiple data streams can be crossly used to maintain the learning performance. But this effectiveness may disappear when the data distribution of streams has significant differences. This is the reason why the results of the augmented training process are relatively lower.

5.3 Evaluate the Selectively Augmented Learning Process

In this experiment, we selectively use the proposed augmented training process and traditional retraining process to reduce the influence of different drift severity of multiple data streams. The same as the previous experiment, synthetic and real-world datasets are chosen for model evaluation, and the results are summarized in Tables 3, 4, 5. The results indicate that the proposed TW-CDM method got higher scores on both synthetic and real-world datasets. And the Friedman test results of TW-CDM$_{\lambda=0.5}$ with other benchmark methods show the efficiency. This process can help reduce the influence of different drift severity in multiple data streams. It got a higher evaluation score by comparing with other methods, but the performance still needs further improvement.

6 Conclusion

This paper focuses on multiple data stream learning, a time-warping-based concept drift learning method has been proposed to help identify and adapt the drift that occurs in multiple data streams. A time warping-based drift identification process has been proposed. Then, augment the training process based on the recognized data region. Finally, a selectively augmented training process is given to enhance the learning ability. The performance has been measured by testing it on synthetic and real-world datasets.

There are also limitations of the proposed method, the issue of asynchronous drift in multiple data streams and the learning strategy in a larger number of data should be further considered. In our future work, we will continue to focus on handling multiple data streams with complex concept drift scenarios. Trying to give a more available strategy to improve the efficiency of multiple data stream learning.

Acknowledgments

This work was supported by the Australian Research Council under Grants DP200100700 and FL190100149.

References

1. Álvarez, V., Mazuelas, S., Lozano, J.A.: Minimax classification under concept drift with multidimensional adaptation and performance guarantees. In: International Conference on Machine Learning. pp. 486–499. PMLR (2022)
2. Baena-Garcıa, M., del Campo-Ávila, J., Fidalgo, R., Bifet, A.: Early drift detection method. In: In 4th International Workshop on Knowledge Discovery from Data Streams. Citeseer (2006)
3. Balzanella, A., Verde, R.: Histogram-based clustering of multiple data streams. Knowledge and Information Systems **62**(1), 203–238 (2020)
4. Bifet, A., Gavalda, R.: Learning from time-changing data with adaptive windowing. In: Proceedings of the 2007 SIAM international conference on data mining. pp. 443–448. SIAM (2007)
5. Chandra, S., Haque, A., Khan, L., Aggarwal, C.: An adaptive framework for multistream classification. In: Proceedings of the 25th ACM International Conference on Information and Knowledge Management. pp. 1181–1190. Indianapolis, IN, USA (2016)
6. Elwell, R., Polikar, R.: Incremental learning of concept drift in nonstationary environments. IEEE Transactions on Neural Networks **22**(10), 1517–1531 (2011)
7. Fang, Z., Lu, J., Liu, F., Xuan, J., Zhang, G.: Open set domain adaptation: Theoretical bound and algorithm. IEEE Transactions on Neural Networks and Learning Systems **32**(10), 4309–4322 (2020)

8. Gama, J., Medas, P., Castillo, G., Rodrigues, P.: Learning with drift detection. In: 17th Brazilian Symposium on Artificial Intelligence. pp. 286–295. Springer (2004)
9. Gama, J., Žliobaitė, I., Bifet, A., Pechenizkiy, M., Bouchachia, A.: A survey on concept drift adaptation. ACM Computing Surveys **46**(4), 44 (2014)
10. Haque, A., Wang, Z., Chandra, S., Dong, B., Khan, L., Hamlen, K.W.: Fusion: An online method for multistream classification. In: Proceedings of the 2017 ACM on Conference on Information and Knowledge Management. pp. 919–928. Singapore (2017)
11. Jaysawal, B.P., Huang, J.W.: Psp-ams: progressive mining of sequential patterns across multiple streams. ACM Transactions on Knowledge Discovery from Data **13**(1), 1–23 (2018)
12. Kolter, J.Z., Maloof, M.A.: Dynamic weighted majority: An ensemble method for drifting concepts. Journal of Machine Learning Research **8**, 2755–2790 (2007)
13. Krawczyk, B., Minku, L.L., Gama, J., Stefanowski, J., Woźniak, M.: Ensemble learning for data stream analysis: A survey. Information Fusion **37**, 132–156 (2017)
14. Li, W., Yang, X., Liu, W., Xia, Y., Bian, J.: Ddg-da: Data distribution generation for predictable concept drift adaptation. In: Proceedings of the AAAI Conference on Artificial Intelligence. pp. 4092–4100 (2022)
15. Liu, A., Lu, J., Liu, F., Zhang, G.: Accumulating regional density dissimilarity for concept drift detection in data streams. Pattern Recognition **76**, 256–272 (2018)
16. Liu, A., Song, Y., Zhang, G., Lu, J.: Regional concept drift detection and density synchronized drift adaptation. In: IJCAI International Joint Conference on Artificial Intelligence (2017)
17. Lu, J., Liu, A., Dong, F., Gu, F., Gama, J., Zhang, G.: Learning under concept drift: A review. IEEE Transactions on Knowledge and Data Engineering **31**(12), 2346–2363 (2018)
18. Lu, J., Niu, L., Zhang, G.: A situation retrieval model for cognitive decision support in digital business ecosystems. IEEE Transactions on Industrial Electronics **60**(3), 1059–1069 (2012)
19. Lu, J., Yang, X., Zhang, G.: Support vector machine-based multi-source multi-attribute information integration for situation assessment. Expert Systems with Applications **34**(2), 1333–1340 (2008)
20. Lu, J., Zuo, H., Zhang, G.: Fuzzy multiple-source transfer learning. IEEE Transactions on Fuzzy Systems **28**(12), 3418–3431 (2019)
21. Mihai, N., Alexandru, M., Bala-Constantin, Z.: Multimodal emotion detection from multiple data streams for improved decision making. Procedia Computer Science **214**, 1082–1089 (2022)
22. Nadeem, A., Verwer, S.: Secleds: Sequence clustering in evolving data streams via multiple medoids and medoid voting. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases. pp. 157–173. Springer (2022)
23. Otero, A., Félix, P., Márquez, D.G., García, C.A., Caffarena, G.: A fault-tolerant clustering algorithm for processing data from multiple streams. Information Sciences **584**, 649–664 (2022)
24. Sangma, J.W., Pal, V., Kumar, N., Kushwaha, R., et al.: Fhc-nds: Fuzzy hierarchical clustering of multiple nominal data streams. IEEE Transactions on Fuzzy Systems **31**(3), 786–798 (2022)
25. Song, Y., Zhang, G., Lu, H., Lu, J.: A fuzzy drift correlation matrix for multiple data stream regression. In: 2020 IEEE International Conference on Fuzzy Systems. pp. 1–6. IEEE (2020)
26. Yu, E., Song, Y., Zhang, G., Lu, J.: Learn-to-adapt: Concept drift adaptation for hybrid multiple streams. Neurocomputing **496**, 121–130 (2022)
27. Yu, H., Liu, W., Lu, J., Wen, Y., Luo, X., Zhang, G.: Detecting group concept drift from multiple data streams. Pattern Recognition **134**, 109113 (2023)
28. Zhou, M., Song, Y., Zhang, G., Zhang, B., Lu, J.: An efficient bayesian neural network for multiple data streams. In: 2021 International Joint Conference on Neural Networks (IJCNN). pp. 1–8. IEEE (2021)